Analyse du comportement d'un Malware Linux avec Kunai et Nushell

Jean-Marc Pouchoulon

Novembre 2025

L'objectif de ce TP est d'analyser un "malware" Linux dans une "sandbox" instrumentée par Kunai. Vous travaillerez individuellement pour mettre en oeuvre la solution kunai-sandbox. Mais vous ferez l'analyse comportementale du malware en binôme.

1 Build de l'Environnement du TP

L'instalation de tous les outils se fera sur une debian13 (ova de l'enseignant) virtualisée sous "Vmware workstation". Cette VM sera configurée avec 16Go de RAM et 4 vCPU avec l'activation de la virtualisation imbriquée (nested virtualization)

1.1 Généralités

Kunai est un outil de monitoring de la sécurité pour Linux. Il utilise la technologie eBPF (Extended Berkeley Packet Filter) permettant une interaction sécurisée avec le kernel Linux. Cette technologie performante permet de capturer les appels systèmes (sycalls) réalisés par les processus en cours d'exécution sur le système hôte Linux.

Kunai peut ainsi être utilisé afin de détecter des comportements malveillants à partir d'indicateurs de compromission (IoC issus par exemple de l'outil "Misp") ou de règles "Yara" lors du dépôt d'un fichier dans une directory.

Les fichiers de traces produits par Kunai sont au format "JSON multilines".

Ils peuvent être lus, convertis dans d'autres formats comme "parquet" (un format "csv" moderne), puis analysés par des outils comme "Nushell" et son plugin "polars" bien adapté à l'analyse de données.

La documentation de Kunai est disponible sur `https://why.kunai.rocks/`

NB: attention à télécharger le pdf du TP depuis la page github afin de rendre les liens actifs de ce document "pdf".

1.2 Pré-requis logiciels

Le TP se fait sur machine virtuelle Debian13 accessible sur ici. Vous utiliserez les outils suivants :

— NuShell qui s'installe sous forme de package debian (normalement l'installation est faite):

```
% curl -fsSL https://apt.fury.io/nushell/gpg.key | sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/fury-nushell.gpg
% echo "deb https://apt.fury.io/nushell/ /" | sudo tee /etc/apt/sources.list.d/fury.list
% sudo apt update
% sudo apt install nushell
```

— nukunai qui est un ensemble de scripts Nushell ayant pour ambition de faciliter l'analyse des traces produites par kunai.
— qemu-kvm pour la virtualisation de la "sandbox".
— kunai que vous pouvez installer depuis la page des releases

2 Premiers pas avec Kunai et Nushell

1. Lancez Kunai sous NuShell.

   Lancez Kunai sous Nushell

   ```
   sudo su -s  /bin/nu
   kunai run --include=all |tee {save -f /tmp/kunai.ndjson}
   CTRL-C
   exit
   ```

2. Nettoyez si nécessaire /tmp/kunai.json pour éliminer le dernier évènement incomplet.
3. Créez et exportez la variable d'environnement NUKUNAI qui pointera sur le répertoire "nukunai" cloné depuis de nukunai.

   ```
   cd ~
   git clone https://github.com/pushou/nukunai.git
   export NUKUNAI='/root/nukunai'
   ```

4. Transformez le fichier de traces au format parquet à l'aide de nukunai:

   ```
   nu $NUKUNAI/kunai_to_parquet.nu  /tmp/kunai.ndjson
   ```

5. Explorez le fichier au format parquet avec Polars :

   ```
   sudo su -s  /bin/nu
   polars open /tmp/kunai.ndjson.parquet|polars into-nu|flatten --all|flatten --all |explore
   ```

6. Affichez la table de tous les évènements Kunai avec "print_events_table.nu"

   ```
   nu $NUKUNAI/kunai_print_events_table.nu
   ```

7. Visualisez les évènements Kunai avec kunai_events_analysis.nu et regardez la documentation des évènements sur `https://why.kunai.rocks/docs/events/` pour comprendre les différents champs listés.

   ```
   nu  $NUKUNAI/kunai\_events\_analysis.nu  /tmp/kunai.ndjson.parquet
   ```

8. A l'aide de Polars sélectionnez la colonne "command_line" et affichez les 10 premières commandes passées les plus fréquentes:

```
polars open /tmp/kunai.ndjson.parquet
    | polars into-nu
    | flatten --all
    | flatten --all
    | get command_line
    | uniq -c
    | sort-by -r count
```

9. A l'aide du scripts filter_events.nu affichez les évènements de type "execveat" et "execve":

```
nu /home/pouchou/Nextcloud/dev/dev_nushell/nukunai/kunai_filter_events.nu ./kunai.jsonl.parquet -e 1,2
```

10. Installez Kunai comme service systemd et bloquez sa désactivation "live". Pourquoi est-ce une configuration conseillée ?

    Faites un snapshot de la VM avant cette étape.

```
sudo ./kunai install --systemd --enable-unit
vim /etc/kunai/config.yaml # modifiez les options selon vos besoins rotation des logs..
systemctl start kunail
sudo systemctl start 00-kunai
```

    Revenez au snapshot avant l'installation.

3 Analyse du comportement d'un "Malware" avec "Kunai Sandox"

3.1 Préparation de la Sandbox Kunai avec ks-sandbox-init

1. Clonez le repository sandbox. Vous utiliserez une machine virtuelle Debian 12 au format qcow2 que vous téléchargerez via le script download-images.sh de kunai-sandbox., préalablement commenté des autres téléchargements afin de gagner du temps.
2. Dans une "directory" "projetm", créez une sandbox nommée "sandbox_prctl" avec la commande ks-sandbox-init.

```
curl -LsSf https://astral.sh/uv/install.sh | sh
uv tool install https://github.com/kunai-project/sandbox.git
mkdir projetm
cd projetm
wget https://raw.githubusercontent.com/kunai-project/sandbox/refs/heads/main/scripts/download-images.sh
chmod +x download-images.sh
./download-images.sh # gardez debian12 commentez les autres téléchargements pour aller plus vit
mkdir sandboxes
export SANDBOXES='/root/projetm/sandboxes'
mkdir -p $SANDBOXES/sandbox1
export LIBGUESTFS_BACKEND=direct LIBGUESTFS_DEBUG=1 LIBGUESTFS_TRACE=1
ks-sandbox-init /root/projetm/images/debian/debian-12-genericcloud-amd64.qcow2 $SANDBOXES/sandbox1 --kunai-bin /usr/bin/kunai
```

3.2 Analyse d'un "Malware" Linux avec kunai-sandbox

1. Clonez le repository NGSOTI du CIRCL Luxembourg sous projetm.

   Ce repository contient des fichiers binaires ELF malveillants !!! Ne rendez pas exécutables les fichiers mis à disposition !!! Ne les exécutez pas sur votre machine virtuelle !!!

```
cd /root/projetm
git clone https://helga.circl.lu/NGSOTI/malware-dataset.git
export VIRUSDIR="/root/projetm/malware-dataset/linux"
```

2. Vérifiez que le malware "22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13" est bien un binaire *statique* au format elf64 via les commandes file et readelf.

```
file ./22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
readelf -a ./22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
readelf -d  ./22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
```

3. Lancez le malware dans la sandbox kunai-sandbox et analysez les traces produites par kunai. Vous pouvez modifiez le temps de vie de la "sandbox" en modifiant le fichier de configuration.

```
cd $SANDBOXES/sandbox1
mkdir -p analyse_virus1
export ANALYSE1="$SANDBOXES/sandbox1/analyse_virus1"
export VIRUS1="$VIRUSDIR/22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13"
kunai-sandbox -c $SANDBOXES/sandbox1/config.yaml -f -o $ANALYSE1 \
        -- $VIRUS1/22e4a57ac560ebe1eff8957906589f4dd5934ee555ebcc0f7ba613b07fad2c13
```

4. Expliquez le fonctionnement de la "sandbox" en faisant un schéma d'architecture que vous montrerez à votre enseignant..
5. Rédigez une analyse du "malware" au travers des traces produites dans $ANALYSE1 par Kunai et Suricata en vous aidant de la première partie du TP. [1]
6. Utilisez suricata pour analyser les traces réseaux du fichier dump.pcap.

```
apt install suricata
suricata-update
mkdir suricata_output
cd suricata_output
suricata -S /var/lib/suricata/rules/suricata.rules -v -r  ../dump.pcap
sudo su -s  /bin/nu
cat eve.json |from json -o |flatten --all|flatten --all|explore
```

7. Présentez en binôme oralement (trois minutes au maximum) ce rapport à votre enseignant au travers d'un petit support. Il sera aussi à rendre en binôme sur Moodle.
8. **Effacez impérativement la machine virtuelle VMware ayant servi au TP.**

1. Vous pouvez vous aider de cet article de la société trendmicro